AMENDMENTS TO THE CLAIMS

The following listing of claims will replace all prior versions and listings of claims in the application.

LISTING OF CLAIMS

1. (Previously Presented) A connection controller for a network comprising a plurality of first stage switches, a plurality of second stage switches coupled to each of the plurality of first stage switches, and a plurality of packet sources coupled to the plurality of first stage switches and each configured to request a traffic pattern for a packet, the connection controller comprising:

a network topology cache capable of being coupled to the network and configured to receive network topology data from the network;

a packing algorithm circuit coupled to the network topology cache and capable of being coupled to the plurality of packet sources, the packing algorithm circuit configured to:

receive the network topology data from the network topology cache,

receive the traffic pattern request from each of the plurality of packet sources in a predetermined time window, and

compute an actual traffic pattern for the packet based on the received network topology data and all the received traffic pattern requests, wherein the actual traffic pattern comprises one of the plurality of first stage switches and one of the plurality of second stage switches such that the network is able to operate as a strictly non-interfering network; and

a logical network state entity coupled to the packing algorithm circuit and capable of being coupled to the packet source, the logical network state entity configured to communicate the computed actual traffic pattern to the source.

2. (Previously Presented) The connection controller of claim 1, wherein the connection controller is configured to calculate a plurality of routing trees including the plurality of first stage and second stage switches, wherein the connection controller calculates a plurality of Destination Location Identifiers (DLIDs) and a set of forwarding instructions for each of the plurality of first stage and second stage switches, wherein the plurality of DLIDs correspond to one of the plurality of routing trees and one of a plurality of destinations in the network, and wherein the connection controller populates a forwarding table of each of the plurality of first stage and second stage_switches with the plurality of DLIDs and the set of forwarding instructions.

3. (Previously Presented) The connection controller of claim 1, wherein computing an actual traffic pattern comprises executing a rearrangement algorithm and assigning one of a plurality of Destination Location Identifiers (DLIDs) to the packet such that the network operates as a strictly non-interfering network.

4. (Previously Presented) The connection controller of claim 3, wherein the packet follows a path through the one of the plurality of first stage switches and the one of the plurality of second stage switches, and wherein the one of the plurality of first stage switches and the one of the plurality of second stage switches forwards the

packet according to the one of the plurality of DLIDs assigned to the packet such that the network operates as a strictly non-interfering network.

5. (Previously Presented) The connection controller of claim 4, wherein the one of the plurality of first stage switches and the one of the plurality of second stage switches each looks up the one of the plurality of DLIDs assigned to the packet in a forwarding table within the one of the plurality of first stage switches and the one of the plurality of second stage switches.

6. (Previously Presented) The connection controller of claim 4, wherein the one of the plurality of first stage switches and the one of the plurality of second stage switches forwards the packet in accordance with the one of the plurality of DLIDs assigned to the packet as found in a forwarding table at each the portion of the plurality of switches.

7. (Previously Presented) The connection controller of claim 1, wherein the network is a CLOS network.

8. (Previously Presented) A connection controller for a network comprising a plurality of first stage switches including a forwarding table, a plurality of second stage switches including a forwarding table and coupled to each of the plurality of first stage switches, and a plurality of nodes coupled the plurality of first stage switches, the connection controller comprising a computer-readable medium containing computer instructions for a processor that when executed by the processor, cause the processor to perform a method comprising the steps of:

calculating a plurality of routing trees, each routing tree comprising the plurality of first stage switches, and one of the plurality of second stage switches;

assigning a Destination Location Identifier (DLID) to each routing tree and to each node;

calculating a set of forwarding instructions for each of the plurality of first stage switches and each of the plurality of second stage switches based on the assigned DLIDs, wherein the set of forwarding instructions causes the one of the plurality of first stage switches and the one of the plurality of second stage switches in each routing tree to operate in a manner that creates a path between each of the plurality of nodes;

populating the forwarding table of each of the plurality of first stage switches and the plurality of second stage switches with the assigned DLIDs and the set of forwarding instructions; and

computing and assigning a DLID to a packet to be transferred in the network based on all traffic pattern requests from the plurality of nodes in a predetermined time window such that the network operates as a strictly non-interfering network.

9. (Previously Presented) The connection controller of claim 8, wherein the network is a CLOS network.

10. (Previously Presented) The connection controller of claim 8, wherein each of the plurality of nodes comprises a destination, and wherein the destination is identified by a BaseLID.

11. (Previously Presented) The connection controller of claim 8, wherein each of the plurality of second stage switches comprises a spine node, and wherein calculating the plurality of routing trees comprises, for each spine node in the network, calculating a shortest path from each spine node to each of the plurality of nodes.

12. (Previously Presented) The connection controller of claim 8, wherein each of the plurality of second stage switches comprises a spine node, and wherein each of the plurality of routing trees further comprises a plurality of links that form a shortest path from each end node to each spine node.

13. (Previously Presented) A computer-readable medium containing computer instructions for instructing a processor to perform a method for forwarding a packet from a source node of a plurality of nodes to a destination node of the plurality of nodes within a network comprising a plurality of first stage switches assigned a respective DLID, and a plurality of second stage switches coupled to each of the plurality of first stage switches and assigned a respective DLID, wherein a first switch of the plurality of first stage switches is further coupled to the source node and a second switch of the plurality of first stage switches is coupled to the destination node, the computer instructions, when executed by the processor, cause the processor to perform a method comprising the steps of:

computing and associating a destination node DLID with the packet based on the destination node and all traffic pattern requests from the plurality of nodes in a predetermined time window such that the network operates as a strictly non-interfering network; and

routing the packet along a path through the first switch of the plurality of first stage switches, one of the plurality of second stage switches, and the second switch of the plurality of first stage switches to the destination based on the first switch DLID, the DLID of the one of the plurality of second stage switches, and the second switch DLID, wherein the first switch of the plurality of first stage switches, the one of the plurality of second stage switches, and the second switch of the plurality of first stage switches forward the packet according to the destination node DLID associated with the packet such that the network operates as a strictly non-interfering network.

14. (Cancelled).

15. (Previously Presented) The connection controller of claim 13, wherein the network is a CLOS network.

16. (Previously Presented) The connection controller of claim 13, wherein routing the packet comprises looking up the destination node DLID associated with the packet in a forwarding table within each of the first switch of the plurality of first stage switches, the one of the plurality of second stage switches, and the second switch of the plurality of first stage switches along the path from the source node to the destination node.

17. (Previously Presented) The connection controller of claim 13, wherein routing step comprises the step of routing the packet in accordance with the destination node DLID associated with the packet as found in a forwarding table included within each of the first switch of the plurality of first stage switches, the one of the plurality of second stage switches, and the second switch of the plurality of first stage switches.

18. (Previously Presented) The connection controller of claim 1, wherein each of the plurality of first stage switches is an INFINIBAND switch and each of the plurality of second stage switches is an INFINIBAND switch.

19. (Previously Presented) The connection controller of claim 8, wherein each of the plurality of first stage switches is an INFINIBAND switch and each of the plurality of second stage switches is an INFINIBAND switch.

20. (Previously Presented) The computer-readable medium of claim 13, wherein each of the plurality of first stage switches is an INFINIBAND switch and each of the plurality of second stage switches is an INFINIBAND switch.

21. (Previously Presented) The computer-readable medium of claim 13, further comprising instructions that, when executed by the processor, cause the processor to further perform the steps of:

recognizing if a new node, a new switch, or both is added to the network; and executing a rearrangement algorithm for the network in response to recognizing the new node, the new switch, or both.